REPORT DOCUMENTATION PAGE

AFRL-SR-BL-TR-98-

0113

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** 30 Jan 97	**3. REPORT TYPE AND DATES COVERED** FINAL TECH RPT, 30 SEP 92 - 29 SEP 96

4. TITLE AND SUBTITLE

A Combined Analytic and Inductive Approach to Learning in Knowledge-based Systems

5. FUNDING NUMBERS

F49620-92-J-0430

6. AUTHOR(S)

Dr. Michael J. Pazzani

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Dept of Information and Computer Science
University of California, Irvine
Irvine CA 92697

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFOSR/NM
110 Duncan Avenue Suite B115
Bolling AFB DC 20332-8050

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT

Unlimited Distribution

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

A new analytic learning algorithm that dynamically evaluates the generality of learned concepts to maximize information gain has been developed.

14. SUBJECT TERMS

machine learning, knowledge-based systems

15. NUMBER OF PAGES

7

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Uncalssified	UL

Final Report: A combined analytic and inductive approach to learning in knowledge-based systems AFOSR Grant F49620-92-J-0430

Michael J. Pazzani
Department of Information and Computer Science.
University of California, Irvine
Irvine, CA 92697
pazzani@ics.uci.edu

Introduction

This project investigated the use of machine learning methods to facilitate the acquisition of classification rules for knowledge-based systems and the maintenance of large knowledge bases. The goal of this research is to develop general purpose methods for learning rules for knowledge-based systems. This learning research was initially focused on learning rules that improve the accuracy of an existing knowledge base. After initial success in this area, attention turned to the problem of learning and revising profiles of user interests for intelligent agent applications.

There are two general approaches to learning classification rules. Inductive learning programs operate by finding regularities among a group of training examples. Analytic learning systems use existing knowledge to explain the classification of examples, and form a general description of the class of examples with the same explanation. In this research, we are investigating an approach to learning classification rules that integrates inductive and analytic learning methods. The goal of this integration is to learn classification rules that are more accurate than the existing knowledge and that are more accurate than rules that would arise from using inductive learning alone. The goal of this learning process may be summarized as follows:

Given: 1. A set of training examples classified as positive and negative examples.
2. (OPTIONAL) An existing knowledge base.
Produce: Classification rules that distinguish positive from negative examples.

Background: Existing state-of-the-art

Prior to this project, analytic learning methods relied upon *operationalization* (Mitchell et al., 1986) to create rules. In operationalization, a depth-first proof of a single positive example is formed. A new rule is formed by conjoining the operational predicates (e.g., the leaves of the proof tree– see Figure 1). A limitation of this form of learning is that it assumed that the existing background knowledge used in the proof process is correct. In FOCL (Pazzani & Kibler, 1991), the operationalization process was enhanced to deal with the possibility that the existing knowledge was incorrect. In particular, FOCL uses a set of positive and negative examples, rather than a single positive example. When there are several ways of operationalizing a literal (i.e., there are multiple, disjunctive clauses), an information-based evaluation function (Quinlan, 1990) is used to determine which clause should be used. Therefore, rather than producing an operationalization from a single positive example, FOCL produces operationalizations that best distinguish positive from negative examples. As a further step toward dealing with incorrect knowledge, FOCL also makes use of an inductive learning process, guided by the same information-based evaluation function, to add additional conditions to operationalizations to further distinguish positive from negative examples or to create new rules.

Research Results: Knowledge-based Systems Maintenance

The primary research result in knowledge-based systems maintenance involved the analysis of situation where operationalization does not result in an improvement in accuracy. This situation occurs when there are few training examples and the existing knowledge is very disjunctive. When a domain theory is highly disjunctive, many operationalizations will be needed to account for each possible combination of rules. However, when there are few examples, it is unlikely that there will be at least one example for each accurate combination of rules. (See Pazzani and Brunk (1993) for more details on this problem and its solution.)

In response to this analysis, we have explored the notion of a frontier (see Figure 2) that allows the learner to dynamically determine the generality of learned rules. A frontier differs from an operationalization in three ways. The frontier represented by those nodes immediately above the line in Figure 2, $b \wedge ((m \wedge n \wedge o) \vee (p \wedge q))$, illustrates these differences:

1. Non-operational predicates (e.g., b) can appear in the frontier.
2. A disjunction of two or more clauses that define a non-operational predicate (e.g., $(m \wedge n \wedge o) \vee (p \wedge q)$) can appear in the frontier.
3. A frontier does not necessarily include all literals in a conjunction (e.g., neither c, nor any specialization of c, appears in the frontier).

We have developed a new analytic learning algorithm for deriving frontiers from existing knowledge. The central difference between this algorithm and previous ones is that it dynamically selects the generality of learned concepts to maximize information gain, rather than learning concepts at a given, fixed level of generality. It accomplishes this by using hill climbing search with a set of operators that derive frontiers of the exiting knowledge.

We have evaluated the results of our research on a large problem from NYNEX (the parent company of New York Telephone and New England Telephone). Nynex Max (Rabinowitz, et al., 1991) is an expert system used at several sites to determine the location of a malfunction for customer-reported telephone troubles. It can be viewed as solving a classification problem where the input is data such as the type of switching equipment, various voltages and resistances and the output is the location to which a repairman should be dispatched (e.g., the problem is in the customer's equipment, the customer's wiring, the cable facilities, or the central office). Nynex Max requires some customization at each site in which it is installed.

We compared the effectiveness of FOCL using frontiers and FOCL using operationalization at customizing the Nynex Max knowledge-base. The existing rules are taken from one site, and the training data is the desired output of Nynex Max at a different site. We repeated 10 runs of each algorithm on 500 randomly selected training examples and evaluated the accuracy on an independent set of 300 test examples. Table 1 shows the accuracy of the two versions of FOCL. For comparison purposes, the results of inductive learning alone and the accuracy of the initial domain theory are also reported. The results indicate that FOCL is more accurate using frontiers than using operationalization. In addition, the frontier results are also significantly more accurate than both the initial knowledge-base or the results of applying inductive learning on the same data.

Table 1: Accuracy at customizing Nynex Max rule base

Condition	Accuracy	95% confidence interval
Operationalization	.938	.01
Frontier	.970	.01
Induction	.917	.02
Initial theory	.946	.01

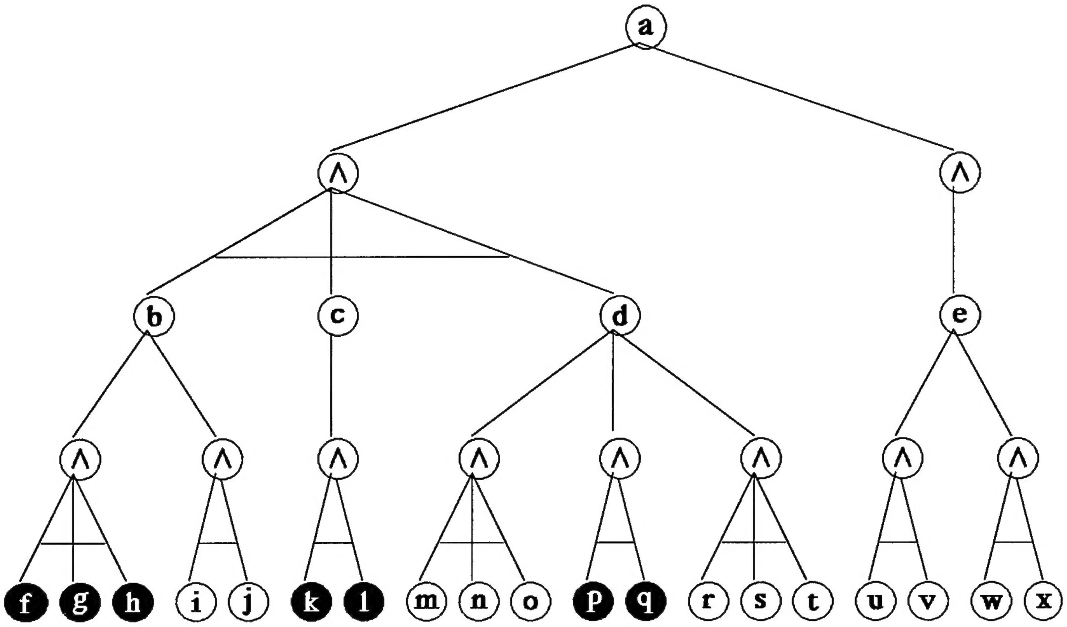


Figure 1: An Operationalization. The bold nodes represent one operationalization (f∧g∧h∧k∧l∧p∧q) of the domain theory. In standard operationalization, this path would be chosen if it were a proof of a single positive example. In FOCL, this path would be taken if the choice made at a disjunctive node had greater information gain (with respect to a set of positive and negative examples) than alternative choices.

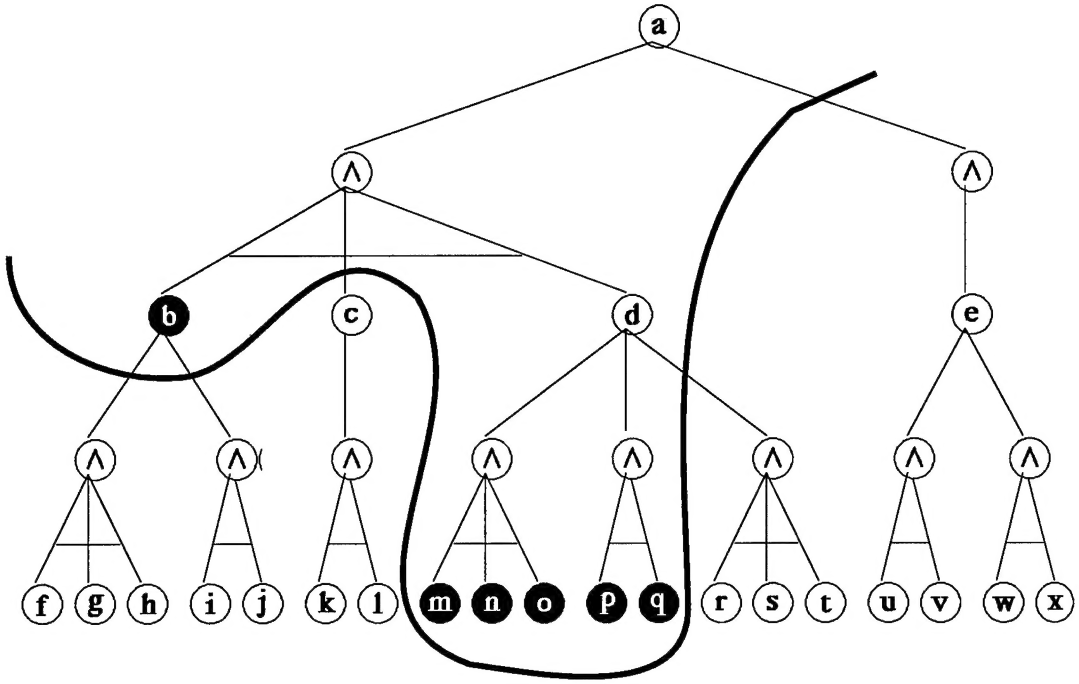


Figure 2: A Frontier. The bold nodes represent one frontier of the domain theory, b∧((m∧n∧o)∨(p∧q)).

Research Results: Learning User Profiles

In the final year of this project, we have focused efforts on learning rules to form a profile of sites on the World Wide Web that a user would like and on the revision of user-defined profiles. This change in focus was motivated by discussions with the ARPA program manager of the Intelligent Integration of Information program to demonstrate the relevance of the technology to the emerging information infrastructure.

We have constructed a software agent that learns to find information on the World Wide Web (WWW), deciding what new pages might interest a user. The agent maintains a separate hot list (for links that were interesting) and cold list (for links that were not interesting) for each topic. By analyzing the information immediately accessible from each link, the agent learns the types of information the user is interested in. This can be used to inform the user when a new interesting page becomes available or to order the users exploration of unseen existing links so that the more promising ones are investigated first.

We are combining statistical algorithms originally developed for information retrieval with machine learning algorithms for learning classification rules. The information retrieval algorithms are used to select keywords that describe the content of pages. The learning algorithms are used to find combinations of these keywords or weightings of the keywords that reliably distinguish interesting pages from noninteresting ones. We have been able to achieve up to 92% accuracy with such combined methods.

Learning algorithms require a set of positive examples of some concepts (such as web pages one is interested in) and negative examples (such as web pages one is not interested in). In this paper, we learn a concept that distinguishes pages rated as hot by the user from other pages (combining the two classes lukewarm and cold, since few pages are rated lukewarm, and we are primarily interested in finding pages a user would consider hot). Most learning programs require that the examples be represented as a set of feature vectors. Therefore, we have constructed a method of converting the HTML source of a web page into a Boolean feature vector. Each feature has a Boolean value that indicates whether a particular "word" is present (at least once) or absent in a particular web page. For the purposes of this paper, a word is a sequence of letters, delimited by nonletters. For example, the URL `<A HREF= http://golgi.harvard.edu/biopages/all.html>` contains nine "words" `a, href, http, golgi, harvard, edu, biopages, all,` and `html`. All words are converted to upper case.

Not all words that appear in an HTML document are used as features. We use an information-based approach, similar to that used by an early version of the NewsWeeder program to determine which words to use as features. Intuitively, one would like words that occur frequently in pages on the hotlist, but infrequently on pages on the coldlist (or vice versa). This is accomplished by finding the expected information gain (*E(W,S)*) (e.g., Quinlan, 1984) that the presence or absence of a word (*W*) gives toward the classification of elements of a set of pages (*S*):

$$E(W,S) = I(S) - [P(W{=}present)I(S_{w=present}) + P(W{=}absent)I(S_{w=absent})]$$

where

$$I(S) = \sum_{c \in \{hot,\ cold\}} -p(S_c)log_2(p(S_c))$$

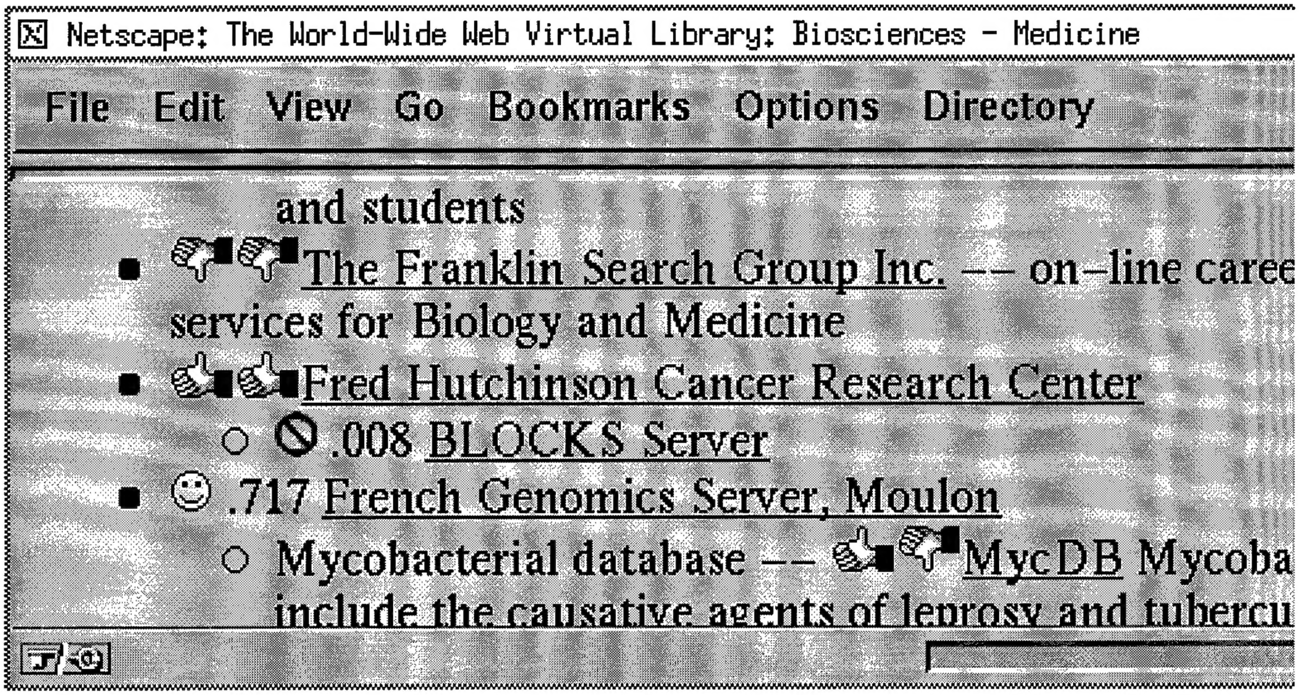
and *P(W=present)* is the probability that *W* is present on a page, and $(S_{w=present})$is the set of pages that contain at least one occurrence of *W* and S_c are the pages that belong to the class *c*.

Using this approach, we find the set of *k* most informative words. Table 2 shows some of the most informative words obtained from a collection of 140 HTML documents on independent rock bands.

Table 2. Some of the words used as features.

nirvana	suite	lo	fi	snailmail	him
pop	records	rockin	little	singles	recruited
july	jams	songwriting	college	rr	his
following	today	write	handling	drums	vocals
island	tribute	previous	smashing	haunting	bass
favorite	airplay	noise	cause	fabulous	becomes

Once the HTML source of items on the hotlist and items on the coldlist for a given topic have been converted to positive and negative examples represented as feature vectors, it's possible to run many learning algorithms on the data. We are particularly interested in those learning algorithms that may be run quickly, so that it would be possible to develop a user profile as the user in browsing. For this reason, we concentrated on Bayesian classifiers for learning a user profile. Once this has been done, we can make suggestions about pages to visit relatively quickly. Each link is annotated with an icon indicating the user's rating or its prediction of the user's rating . Two thumbs up indicates the user has visited the page and rated it hot, one thumb up and one down indicates a previous lukewarm rating, and two thumbs down indicates a previous cold rating. A smiley face indicates that the user hasn't visited the page and it is predicted that the user would like the page. the page to the user. The international symbol for "no" is used to indicate the page hasn't been visited and the learned user profile indicates the page should be avoided. Following any prediction is a number between 0 and 1 indicating the probability the user would like the page. Note that these ratings and predictions are specific to one user and do not reflect on how other users might rate the pages.



☒ Netscape: The World-Wide Web Virtual Library: Biosciences - Medicine
File
Edit
View
Go
Bookmarks
Options
Directory

and students

- 👎👎The Franklin Search Group Inc. -- on-line caree services for Biology and Medicine
- 👍👍Fred Hutchinson Cancer Research Center
  - ○ ⊘ .008 BLOCKS Server
- ☺ .717 French Genomics Server, Moulon
  - ○ Mycobacterial database -- 👍👎MycDB Mycoba include the causative agents of leprosy and tubercu

Conclusions

We have completed all of the major goals:

1. A learning system was developed that learns rules from data and can optionally accept as input existing rules, lexical knowledge, and misclassification costs. With this addition knowledge, the rules are optimized to be similar to existing rules, be understandable to people, and to minimize the cost of errors.

2. A related learning system was developed that can classify text and was evaluated on learning and revising user profiles for sites on the WWW.

This new learning strategies reflects a different bias than previous approaches in that existing knowledge is used unless explicitly contradicted by training data. An application of the new approaches to learning developed under this grant yielded promising results on customizing a large operational knowledge-based system and are now being deployed in a publicly available intelligent agent.

References

Ali, K. & Pazzani, M. (1993). HYDRA: A noise-tolerant relational concept learning algorithm. To appear in the *Proceedings of the International Joint Conference on Artificial Intelligence*, Chambery, France.

Ali K. and Pazzani M. (1995) HYDRA-MM: *Learning Multiple Descriptions to Improve Classification Accuracy*. International Journal on Artificial Intelligence Tools, 4.

Ali K. and Pazzani M. (1996). *Error Reduction through Learning Multiple Descriptions* Machine Learning, 24:3.

Domingos, P., and Pazzani, M. (in press). Beyond Independence: Conditions for the Optimality of the Simple Bayesian Classifier. *Machine Learning*.

Merz, C. Pazzani, M, & Danyluk, A. (1996). Tuning Numeric Parameters to troubleshoot a telephone network. IEEE EXPERT, Feb 1996, pg. 44-49.

Mitchell, T., Keller, R., & Kedar-Cabelli, S. (1986). Explanation-based learning: A unifying view. *Machine Learning, 1,* 47–80.

Pazzani, M., & Kibler, D. (1992). The role of prior knowledge in inductive learning. *Machine Learning, 9,* 57-94.

Pazzani, M. & Brunk, C. (1993) Finding accurate frontiers: A knowledge-intensive approach to relational learning. The *Proceedings of the National Conference on Artificial Intelligence.* Washington, D.C.

Pazzani M. and Billsus, D. (in press). *Learning and Revising User Profiles: The identification of interesting web sites* Machine Learning.

Quinlan, J.R., (1990). Learning logical definitions from relations. *Machine Learning, 5,* 239–266.

Rabinowitz, H. et al. (1991). Nynex Max: A telephone trouble screening expert system. In R. Smith & C. Scott (Eds.) *Innovative applications of artificial intelligence, 3,* 213–230.